

## Trabalho de Programação Orientada a Objeto

Implementar as classes Contato, ContatoPessoal e ContatoProfissional. As classes ContatoPessoal e ContatoProfissional são subclasses da classe Contato. A classe Contato deve ser declarada como abstrata. Os atributos e métodos de cada uma dessas classes são mostrados a seguir:

- a. Classe Contato:
  - Atributos: int id; String nome; String endereço; String email; String telefone
  - Métodos:
    - . Construtor – para atualizar todos os atributos da classe
    - . Construtor – não atualizar apenas o email
    - . get/set – para cada um dos atributos
    - . void mostraDetalheContato ( ) – método abstrato
- b. Classe ContatoPessoal:
  - Atributos: String dataAniversario; String relacaoDeProximidade
  - Métodos:
    - . Construtor – para atualizar todos os atributos da classe
    - . Construtor – não atualizar apenas o email
    - . get/set – para cada um dos atributos
    - . void mostraDetalheContato ( ) – mostrar todos os atributos da classe
- c. Classe ContatoProfissional:
  - Atributos: String nomeEmpresa; String cargo
  - Métodos:
    - . Construtor – para atualizar todos os atributos da classe
    - . Construtor – não atualizar apenas o email
    - . get/set – para cada um dos atributos
    - . void mostraDetalheContato ( ) – mostrar todos os atributos da classe

Implementar uma classe denominada de Agenda com os seguintes atributos e métodos:

- Atributos: declarar um vetor de tamanho 100 para armazenar objetos do tipo Contato.
- Método:
  - . void insertContato (Contato c) – esse método adiciona o objeto c recebido como parâmetro em uma posição livre do vetor. Antes de inserir, verificar se existe algum contato cadastrado que tenha o atributo id igual ao atributo id do contato passado como parâmetro. Se existir mostrar a mensagem indicando que o contato já está cadastrado
  - . Contato buscarContato(int id) – esse método retorna o contato cadastrado no vetor que tenha o id igual ao parâmetro id informado. Se não encontrar, retornar null.
  - . Contato deleteContato(int id) – esse método percorre o vetor até encontrar um contato que tenha o atributo id igual ao parâmetro id informado. Se encontrar, o contato deve ser retirado

do vetor e retornado pelo método. Se não encontrar, mostrar mensagem explicativa e retorna null.

. void updateContato(Contato c) – esse método percorre o vetor até encontrar um contato que tenha o atributo id igual ao id do parâmetro informado. Se encontrar, verificar se o objeto cadastrado no vetor é da mesma classe do objeto passado como parâmetro. Se for da mesma classe, fazer a alteração. Se não for mostrar mensagem explicativa. Se não encontrar o id procurado, mostrar a mensagem que o contato não foi encontrado.

. ContatoPessoal [] – obterContatoPessoal () – esse método retorna um vetor com todos os objetos do tipo ContatoPessoal cadastrados no vetor.

. ContatoProfissional [] – obterContatoProfissional () – esse método retorna um vetor com todos os objetos do tipo ContatoProfissional cadastrados no vetor.

. ContatoPessoal [] – obterAniversariante (String data) – esse método retorna um vetor com todos os objetos do tipo ContatoPessoal que tenham o dia e mês do atributo dataAniversario igual ao dia e mês do parâmetro data informado.